

InstaVine

Manan Vyas, Nicholas Yingst, Shruti Sharma

February 10, 2014

Abstract

The Instagram app went viral soon after its launch. It became so popular that Facebook decided to take over it. The motivation for this project is to take the features of Instagram one step closer and create a real-time video Instagram. The user will be able to choose from a given set of filters that can be applied to the video that is being captured by the camera in real-time. The chosen filter will be applied to the captured video and displayed on the monitor.

Description of the system

The system will consist of the following parts:

1. Camera for capturing the video
2. DSP board
 - the switches on the board will be used for choosing the filter to be applied
 - the DSP chip will perform the necessary processing to apply the filter to each frame of the video (24frames/second)
3. Monitor for displaying the real-time video with the filter applied

The advantage of this project over the already existing apps is that the effect of the filter can be seen as the video is being captured. In the existing apps, the filters are applied on captured and stored images or videos. A possible application of this project can be to integrate these features into digital cameras. This will allow the user to instantly capture videos with filters already applied to them.

Description of Possible Algorithms

Effect: Pencil sketch

Algorithm: Edge detection using adaptive canny

Description: The threshold for edges is computed by gradient histogram difference diagram. The image is then classified by the complexity of its edge information. Based on the edge information, the images are classified as little /rich/numerous edge information images. Different formulae are used for calculating the high and low thresholds for each type of image.[1][5]

Effect: Oil painting

Algorithm: Mean shift segmentation

Description: Using the mean-shift algorithm, the clusters in each frame are identified. The clusters all assume the color of the mean color value of that segment. This results in smoother colors across the image and gives an oil painting like effect.[4][10]

Effect: Contrast enhancement

Algorithm: Histogram equalization

Description: The histograms of the three channels of the frame are computed. The colors of each pixel of the image are then adjusted such that the resulting histograms are spread out more evenly throughout the range of that channel (0-255). This increases the global contrast of the image and makes the image quality better.[12]

Effect: Applying different filters to the foreground and the background (for ex: foreground color and background black and white, etc.)

Algorithm 1: Foreground-background separation using edge detection

Description: Edge detection is performed and the sharp edges are differentiated from the broken and blurry ones. The sharpest edges will be the ones that will be the nearest to the camera and hence will be classified as the foreground.[8]

Algorithm 2: Foreground-background separation using motion detection

Description: The foreground is assumed to be the moving object. When we subtract one frame from the other of the video, we can detect the moving object and classify that as the foreground. However, we cannot use this because it will not give accurate results for our demo set-up as there will be many moving objects (people) in each frame even in the background.

Effect: Vignette

Algorithm 1: Foreground tracking

Description: Foreground-background separation and tracking the movement of the foreground over the different frames.

Algorithm 2: Adaptive mean-shift for automated object tracking

Description: Background separation followed by shadow elimination is applied as part of pre-processing. Connected component analysis is used to classify objects as objects to track or noise. Then, mean-shift tracking is applied. This includes update condition, re-initialization of mean-shift trackers, object matching and standard mean-shift tracking with masking the search area is applied to all objects in the video.[9][11][7]

Effect: Comic book effect

Algorithm: Dithering and Color Half toning

Description: Similar to the color half toning effect that can be added in Photoshop, the comic book effect involves using error diffusion techniques like Floyd Steinberg, Stucki etc. to dither the RGB frames of the video.[3][2]

Effect: White balance correction

Description: Accurate colors can be difficult to map inside due to lights with differing color temperatures and non-uniform lighting. In order to correct the white balance a known white and black/grey object (eg. Sheet of paper with black and white) can be placed in the field of view and each color channel histogram can be remapped to display each color correctly.[6]

Complexity Analysis

1. Edge detection using Adaptive Canny Technique

The canny edge detection algorithm consists of the following steps:

- Convolution of image with the Gaussian blur kernel to reduce noise in image as the canny edge detector is susceptible to noise degradation.
- Convolution of the image with the four Canny edge detector kernel.
- Computation of gradient magnitude and gradient direction.
- Thresholding so that the image is said to consist of 10-15 percent of the total pixels as edge pixels containing edges.

For an image of dimension $M \times N$; convolutions can be efficiently performed with the time complexity as $O(MN \log(MN))$ which is the key bottleneck for Canny Edge Detection in our project.

2. Mean Shift filters

It is one of the clustering techniques used in image segmentation. For each pixel within the image, the mean shift filter defines a kernel (usually weighted by a Gaussian kernel) around the pixel of interest and computes the mean of the data. The algorithm then iteratively shift the centroid of the kernel towards the calculated mean.

Calculating the kernel mean is the most computationally expensive step with a time complexity of $O((MN)^2)$ for an $M \times N$ kernel.

3. Dithering

It is generally applied to BW images. For the comic book effect, dithering has to be applied on RGB channels separately (this can also go in Challenges section). However, the dithering algorithm is comparatively computationally inexpensive in time $O(M \times N)$ as compared to other algorithms like mean shift filter.

Major Challenges

- Effectively utilize the memory on DAVINCI video board since it is limited.
- 24 fps for processing purposes.
- Find a way to store the captured and processed video to external memory for storage purposes as a novel addition to the system.
- Optimize the code for implementing algorithms for various filters discussed so there is minimum lag in the system.

Human Factors

User controls are not a primary function of this project. If time permits, a graphical user interface will be created, however the primary form of filter selection will be the onboard DIP switches. The systems success is in large part determined by the users perceived visual quality of the displayed images. Due to the difficulty of determining perceived quality, trials will be conducted to determine the optimal settings for each filter. Users will also have the optional ability to save recorded video outputs onto the connected PC.

Rough schedule

ID	Task Name	Start	Finish	Duration	Feb 2014				Mar 2014				Apr 2014			
					2/9	2/16	2/23	3/2	3/9	3/16	3/23	3/30	4/6	4/13	4/20	4/27
1	Set Project Requirements	2/10/2014	2/14/2014	1w												
2	Write First Iteration	2/17/2014	3/28/2014	6w												
3	Debug	3/28/2014	4/10/2014	2w												
4	Optimize run-time	4/10/2014	4/16/2014	1w												
5	Visual Trials	4/16/2014	4/22/2014	1w												
6	Extend Functionality	4/22/2014	5/2/2014	1.8w												

Final Test Set-Up

System functionality can be easily verified by viewing the image outputs on an external monitor. Filters will be visually confirmed individually and combined. The full system will include an external camera, monitor, PC, and the DSP board. There will be minimal restrictions on the placement of the camera and user, however the user will be expected to remain within the focal area of the camera.

Board

The DM6437 board will be used. Its larger external memory and higher quality video input/output will be necessary to make this project successful. An external monitor and standard definition camera will be used in conjunction with the board.

References

- [1] Y.; Cetin A.E. Akarun, L.; Yardunci. Canny edge detection codec using vlib on davinci series dsp. In *International Conference on Computer Science and Service System (CSSS)*, pages 221,224, 11-13 August 2012.
- [2] Y.; Cetin A.E. Akarun, L.; Yardunci. Adaptive methods for dithering color images. In *IEEE Transactions on IEEE*, vol.6, no.7, pages 950,955, Jan 1997.
- [3] A.U.; Allebach J.P. Baqai, F.A.; Je-Ho Lee; Agar. Digital color halftoning. In *Signal Processing Magazine, IEEE*, vol. 22, pages 87,96, Jan 2005.
- [4] K. J. Sungkuk Chun. Oil painting rendering changeable by light effect. Soongsil University, HCI Lab.
- [5] Luo Tao; Xi feng Zheng; Ding Tie-fu. Self-adaptive threshold canny operator in color image edge detection. In *2nd International Congress on Image and Signal Processing, CISP '09*, pages 1,4, 17-19 October 2009.
- [6] Theo; Weijer Joost V.D. Gijzenij, Arjan; Gevers. Computational color constancy: Survey and experiments. In *IEEE Transactions On Image Processing*, 2010.
- [7] V. Gu, I.Y.-H.; Gui. Colour image segmentation using adaptive mean shift filters. In *International Conference on Image Processing*, vol.1, pages 726,729, 2001.
- [8] Liu Junqiang; Gao Jianming; Shen Qingming; Tian Junwei. A background separation method of nonuniform image segmentation. In *4th IEEE Conference on Industrial Electronics and Applications, ICIEA 2009.*, pages 3049,3053, May 2009.
- [9] L.; Zhang D.; Wu C. Ning, J.; Zhang. Scale and orientation adaptive mean shift tracking. In *Computer Vision, IET*, vol.6, no.1, pages 52,61, January 2012.
- [10] Youngmin; Hong Jeong Park, Sungchan; Ha. A parallel and memory-efficient mean shift filter on a regular graph. In *The 2007 International Conference on Intelligent Pervasive Computing*, pages 254,259, Oct 2007.
- [11] N.; Niblack W. Qian Huang; Dom, B.; Megiddo. Segmenting and representing background in color images. In *Proceedings of the 13th International Conference on Pattern Recognition*, pages 13,17, August 1996.
- [12] Heechul Han; Kwanghoon Sohn. Automatic illumination and color compensation using mean shift and sigma filter. In *2nd International Congress on Image and Signal Processing, CISP '09*, vol.55, no.3, pages 978,986, August 2009.